**Pogledi**

Relacioni sistemi podržavaju poglede – imenovane relvare. Vrednost pogleda je rezultat izvršavanja određenog relacionog izraza u tom trenutku.

Relacioni izraz se navodi pri formiranju pogleda. Sistem pretvara upit naveden pri formiranju pogleda u ekvivalentan upit nad osnovnim relvarima. Pretvaranje se vrši supstitucijom. Supstitucija je moguća na osnovu osobine relacionog zatvorenja.

Primeri:

VAR UPISANI2010 VIEW

(DOSIJE WHERE INDEKS/10000 = 2010) {ALL BUT MESTO\_RODJENJA}

RENAME GOD\_RODJENJA AS GODINA

Pogled UPISANI2010 ima atribute Indeks, Ime, Prezime, Godina

VAR PAR\_NAZIVA VIEW

( (PREDMET RENAME NAZIV AS PNAZIV)

JOIN ISPIT JOIN

(ISPITNI\_ROK RENAME NAZIV AS INAZIV)

) {PNAZIV, INAZIV}

Pogled koji sadrži parove naziva roka i predmeta u kojima je je predmet polagan

Pogled može da bude definisan i nad drugim pogledom, a ne samo nad osnovnim relvarom.

VAR UPISANI\_2010\_1992 VIEW

UPISANI\_2010 WHERE GODINA > 1992

Studenti koji su upisani 2010. godine a rođeni posle 1992.

Formiranje pogleda

Sintaksa naredbe za kreiranje:

VAR <ime relvara> VIEW <relacioni izraz>

<lista kandidata za kljuceve>

[RESTRICT/CASCADE]

<lista kandidata za kljuceve> može biti i prazna ako pogled može da nasledi kandidate za ključeve

RESTRICT/CASCADE može, ali ne mora da postoji

Sintaksa naredbe za brisanje:

DROP VAR <ime relvara>

**Osobine pogleda**

Definicija pogleda kombinuje

* Spoljašnju shemu
* Preslikavanje između konceptualnog nivoa (sadrži izgled spoljašnjeg objekta i opis kako se on preslikava na konceptualni nivo)
* Spoljašnje/spoljašnje preslikavanje (npr. pogled UPISANI\_2010\_1992)

**Funkcije pogleda**

* Obezbeđuju automatsku zaštitu za skrivene podatke
* Omogućuju da različiti korisnici (istovremeno) vide iste podatke na različite načine
* Uprošćavaju složene operacije
* Omogućuju logičku nezavisnost podataka

**Logičkna nezavisnost podataka**

Proširenje relacija baze ne sme da ima efekat na izvršavanje aplikativnih programa

Restruktuiranje baze ne sme da ima efekat na postojeće aplikativne programe. Nova i stara baza treba da budu informaciono ekvivalentne.

Primer restruktuiranja baze: ako se relacija DOSIJE razbije na dve osnovne relacije

VAR DOSIJE1 BASE RELATION

{INDEKS INDEKS, IME IME, PREZIME PREZIME, GOD\_RODJENJA INTEGER}

PRIMARY KEY {INDEKS}

VAR DOSIJE2 BASE RELATION

{INDEKS INDEKS, MESTO\_RODJENJA CHAR}

PRIMARY KEY {INDEKS}

DOSIJE može da se formira kao pogled:

VAR DOSIJE VIEW DOSIJE1 JOIN DOSIJE2

Aplikativni programi koji su radili sa osnovnim relvarom DOSIJE bi trebalo da mogu bez izmena da se referišu na pogled DOSIJE

**Čitanje preko pogleda**

Operacija čitanja podataka preko pogleda se konvertuje u ekvivalentnu operaciju nad osnovnim relvarima

* materijalizacija relacije koja je trenutna vrednost pogleda
* supstitucija relacionog izraza u drugom relacionom izrazu

Semantika pogleda se definiše preko materijalizacije relacija!

**Ograničenja integriteta kod pogleda**

Zlatno pravilo se primenjuje i na poglede, tj. ažuriranje pogleda ne sme da naruši ograničenja integriteta nad pogledima.

Ograničenja integriteta nad pogledaim su izvedena iz ograničenja integriteta osnovnih relvara.

Ažuriranje pogleda

Ako je D baza, V pogled nad bazom D i X funkcija kojom se definiše pogled V, tada je za dati pogled V = X(D) i operaciju ažuriranja pogleda U nad V potrebno odrediti operaciju ažuriranja U1 nad D tako da važi: U(X(D)) = X(U1(D)).

Moguće je odabrati više operacija U1. Koju odabrati?

Codd-ov pristup: definisanje pogleda koji mogu da se ažuriraju

Date-in pristup: svi pogledi mogu da se ažuriraju. Operacije ažuriranja se izvode izbegavanjem ograničenja integriteta u međukoracima ažuriranja. Ažuriranje se izvodi kao brisanje postojećih i unošenje novih podataka.

SQL Podrška

CREATE VIEW naredba

CREATE [OR REPLACE] VIEW

<ime pogleda> AS <izraz nad tabelom>

[WITH [<kvalifikator>] CHECK OPTION]

DROP VIEW <ime pogelda>

Izdvajanje rezultata

Po standardu bi sve operacije sa čitanjem podatak preko pogleda trebalo da rade korektno. Međutim, u praksi to nije slučaj, pogotovo ako se pogledi ne materijalizuju.

Ažuriranje pogleda

SQL/92 podrška za ažuriranje pogleda je vrlo ograničena. Jedini pogledi koji se smatraju mogućim za ažuriranje su pogledi izvedeni iz jedne osnovne tabele kombinacijom restrikcije i projekcije.

U SQL/92 pogled može da se ažurira ako važi:

* Izraz kojim se definiše pogled je SELECT izraz koji ne sadrži JOIN, UNION, INTERSECT ili EXCEPT
* SELECT klauzula ne sadrži ključnu reč DISTINCT
* Svaka SELECT stavka sadrži ime koje predstavlja referencu na kolonu osnovne tabele
* FROM klauzula sadrži referencu na tačno jednu tabelu koja je ili osnovna tabela ili pogled koji može da se ažurira
* WHERE klazula SELECT izraza ne sadrži podupit u kom se FROM klauzula referiše na istu tablu kao i FROM klauzula u SELECT izrazu na najvišem nivou
* SELECT izraz ne sadrži GROUP BY, niti HAVING klauzulu